Java – OOP - Classes and Objects

Presented by

Sagar
Java Consultant

Problem Solving

- The key to designing a software solution is breaking it down into manageable chunks
- •When writing software, we design separate chunks that are responsible for certain parts of the solution
- •An object-oriented approach lends itself to this kind of solution decomposition
- •We will dissect our solutions into chunks called objects and classes

Classes and Objects

- Java is an object-oriented programming language
- As the term implies, an object is a fundamental entity in a Java program
- Objects can be used effectively to represent real-world entities i.e., they are adapted into OOProgramming
- Eg. Each employee object handles the processing and data management related to that employee

Objects

- An object has:
  - –state - descriptive characteristics
  - –behaviors - what it can do
- The state of a bank account includes its account number and its current balance
- The behaviors associated with a bank account include the ability to make deposits and withdrawals
- Note that the behavior of an object might change its state
  - (update on balance)

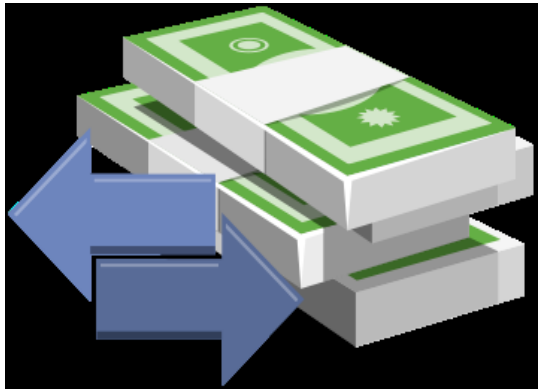
State and Behavior

State

balance

engine,wheels,fuel capacity

Behavior



deposit and withdraw

move, rotate

Classes

- An object is defined by a *class*
- A class is the blueprint of an object
- The class uses methods to define the behaviors of the object
- A class represents a concept, and an object represents the embodiment of that concept
- Multiple objects can be created from the same class

Example

```
public class BankAccount{
  private int accountNumber;
  private string accountHolder;
  private int balance;
  public BankAccount(int acntNo,String acntHldrName ,int bal) {
        accountNumber = acntNo;
        accountHolder = acntHldrName;
        balance = bal;
  }
  public void withdraw(int amount) {
        if (balance > amount)
        balance = balance - amount;
  }

  public void deposit(int amount) {
        balance = balance + amount;
  }
  public void showBalance() {
        System.out.println("Balance = " + balance);
  }
}
```

Example ..

```
public class MainBankAccount{
    public static void main(String[] args) {
            BankAccount sbAcnt = new BankAccount();

             sbAcnt.deposit(7000);
            sbAcnt.withDraw(2000);
            sbAcnt.showBalance();

            BankAccount rdAcnt = new BankAccount();

             rdAcnt.deposit(7000);
            rdAcnt.withdraw(1000);
            rdAcnt.showBalance();
    }
}
```


return
THANK
YOU
ctrl
control